



Mining Urban Traffic Condition from Crowd-Sourced Data

Ha Mai-Tan^{1,3} · Hoang-Nam Pham-Nguyen^{1,3} · Nguyen Xuan Long^{2,3} · Quang Tran Minh^{1,3}

Received: 13 April 2020 / Accepted: 24 June 2020 / Published online: 3 July 2020
© Springer Nature Singapore Pte Ltd 2020

Abstract

Traffic congestion is an inherent and hard issue to be tackled in huge urban areas, particularly in developing countries where transportation infrastructures have not been grown well to fulfill speedy developing request demands. This paper proposes novel solutions to these issues by devising mobile crowd-sourcing based approaches to traffic estimation. A framework for effective collecting, integrating and analyzing traffic-related data shared by mobile crowds has been devised. Besides, essential issues on predicting traffic conditions at streets where real-time data is missed are also resolved by applying data mining techniques to historical data. A prototype system has been developed to validate the proposed solutions. The experimental results show the feasibility and the effectiveness of the proposed methods revealing that they are ready to be applied in the practice.

Keywords Urban traffic condition · Crowd-sourced data · Data mining

Introduction

Traffic congestion is one of the inherent challenges for large cities as it causes not only economic loss but also pollution (e.g., air, noise pollution) and other social issues, declining the quality of life and even public safety [1, 2]. This issue is

more serious in developing countries where transportation infrastructures have not been developed well to satisfy quick-growing demands [3]. Consequently, traffic estimation, monitoring, and prediction to mitigate traffic congestion become highly prioritized issues to be resolved in large cities. In addition, intelligent transportation systems/services (ITS) [4, 5] including traffic estimation and management become main focuses in realizing smart city environment/eco-system in big cities such as in Ho Chi Minh City (HCMC) and Hanoi, Vietnam.

Traffic condition estimation requires updated (in time) and accurate information about traffic-related status on every road/street in the city which is transferred to the system at the center (a server system or data centers on the cloud) for analysis and disseminating of useful traffic information to commuters which helps them avoid traveling to the congested locations. To reach this target, adequate data, e.g., time, location (i.e., at what road segment), congestion level (can be defined in advanced with some principles) or the velocity and the density of the traffic flow, reason that causes the congestion, prediction when the congestion will be cleared, images about the congested segment, should be properly collected and analyzed. In the conventional system, these data are collected by fixed sensor systems deployed on the streets such as loop detector, RFID, camera systems, and so on [1, 3, 6, 7]. Obviously, it is impossible to deploy these systems everywhere around a large city because of

This article is part of the topical collection “Software Technology and Its Enabling Computing Platforms” guest edited by Lam-Son Lê and Michel Toulouse.

✉ Quang Tran Minh
quangtran@hcmut.edu.vn

Ha Mai-Tan
1870303@hcmut.edu.vn

Hoang-Nam Pham-Nguyen
1880316@hcmut.edu.vn

Nguyen Xuan Long
nxlong@hcmut.edu.vn

¹ Department of Information Systems, Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet, District 10, Ho Chi Minh City, Vietnam

² Department of Bridge and Highway Engineering, Faculty of Civil Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet, District 10, Ho Chi Minh City, Vietnam

³ Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam

their high deployment and maintenance costs. Consequently, the coverage of the whole system becomes significantly limited. Moreover, as discussed before, the traffic infrastructures have not yet developed well in developing countries, hence these IoT-based systems are not available, leading to essential issues on data fusion for ITS services [1–3].

Therefore, it is critical to propose appropriate approaches to leveraging multiple crowd-based data sources including fixed-IoT and human-sensing systems to collect timely, on-site and adequate traffic data. Our previous work in [2] has proposed a framework for crowd-sourcing based traffic condition estimation where data from multiple sources such as human-sensed data (e.g., data shared by mobile users via mobile devices), and data from available fixed-sensor systems can be efficiently integrated for traffic estimation. However, the work is still limited in a general framework, where details on analysis of data from mobile users, and resolving the essential issues of missing real-time data are missed.

This current work is an extension of the fundamental work in [2] with following main contributions:

- (i) We present in details an effective mechanism for collecting, integrating and analyzing traffic-related data shared by mobile crowds which commonly contains noises, then estimates traffic condition. The proposed mechanism is lightweight and appropriate to road segment level, hence increasing the accuracy as well as the usefulness of the estimated information.
- (ii) We devise an appropriate solution to the issues of estimating traffic condition at road segments where real-time data is missed. Several data classification models have been examined with historical data to find out the best suited one. Details on historical data integration from both internal and external systems are also presented.
- (iii) A prototype system for real world applications of ITS services, especially for traffic condition estimation and notification in HCMC, has been developed. The evaluation results from experiments using the developed prototype and external data sources have been thoroughly analyzed to confirm the feasibility and the effectiveness of the proposed approaches revealing that they are ready to be applied in the practice.

The rest of the paper is organized as follows: Section "Related Work" reviews the related work. The overall architecture and problem formulation are presented in Sect. "Overall Architecture and Problem Definition". Section "Urban Traffic Estimation from Crowd-Sourced Data" describes the proposed approach, while Section "Prototyping" presents the prototype system. Section "Evaluation" presents and analyzes the evaluation results, and Section "Conclusion" concludes this paper.

Related Work

In [8], Estellés-Arolas et al. survey the scholarly literature on crowd-sourcing and find nearly forty different interpretations of crowd-sourcing, with some scholars using different and conflicting definitions for the term even within their own works. The advances of mobile technologies have enabled mobile crowd-sourcing in various domains including production, service provision, data sharing [9, 10]. Crowd-sourcing are often explained as an act by an organization, a company or persons (i.e., requesters) issuing requests (e.g., to complete tasks or services, to gather adequate data for knowledge discovery, etc.) to others via the Internet [10]. The crowd provides solutions, data or knowledge to the requesters for their original purposes. In contrast, service or data providers are rewarded in accordance with their contributions by various means such as money, social acknowledgment, rights to access information or to use services freed from charge. Crowd-sourcing approaches have been widely employed in disaster management and information sharing in companies or communities [11]. Ushahi could be a successful example of a crowd-sourcing approach for sharing and analyzing data, disseminating evacuation instructions to on-site users and to assist volunteers conducting effective rescues during the Haiti earthquake [12].

Traffic congestion estimation and monitoring have attracted tremendous research interest recently. Various research focuses on systematic processes for data collection, processing and knowledge dissemination to citizens and authorities. Existing traffic status estimation systems such as the VICS (Vehicle Information and Communication System) in Japan [7] majorly depend on road-side fixed sensor systems (e.g., camera systems, RFID, loop detectors, etc.). Basing on various movable things on roads, governments want to monitor traffic status in cities. In [13], Nguyen et al. propose a traffic congestion monitoring system by Internet of Vehicles, which includes data collection, segmented structure establishment, traffic-flow modelling, local segment traffic congestion prediction, and origin-destination traffic congestion service for drivers. In [14], Lewandowski et al. propose a traffic congestion monitoring system by analyzing strength of radio signal received by mobile devices from beacons that are placed on opposite sides of a road. In [15], Aissaoui et al. propose a new real-time traffic monitoring based on emerging vehicular communication system by cluster-based V2X traffic data collection mechanism. In [16], Elloumi et al. propose a road traffic monitoring system using multiple Unmanned Aerial Vehicles, which is an attractive solution for road traffic monitoring. In [17], de Almeida et al. propose a system that monitors and disseminates

traffic conditions using a decentralized infrastructure (DOCTraMS) with applied traffic data from São Paulo, Brazil. In [18], Pham et al. propose a framework for collecting and processing traffic data from various types of sources; and adapting several algorithms in transportation research to regulate the traffic and inform users via a control center in Ho Chi Minh City, Vietnam. These systems are, however, limited in coverage and, more importantly, they are not matured enough in developing countries. To overcome these difficulties, several studies have utilized mobile devices as traffic probes for collecting road traffic data [3, 5, 7, 19]. This approach leverages the ubiquity of mobile phones for improving the coverage when collecting the location of vehicles/commuters via GPS modules on mobile devices. This mechanism is cost-effective in terms of saving costs for infrastructure deployment on the roads. However, processing GPS data to yield traffic status such as traffic congestion requires sophisticated data processing mechanisms as GPS data are semantically far from traffic information while GPS data may contain noises.

For traffic monitoring and congestion mitigation, crowds provide on-site traffic-related data such as location, traffic flow velocity, reasons causing such congestion, the queue length (e.g., how many meters in the front and behind), is authority available for solving traffic congestion, pictures about traffic flow, and so on such as those on Urban Traffic Program provided by existed ITS named CityDrive [20], GreenDrive [21]. An approach to urban traffic condition estimation by aggregating GPS data [22], Ma et al. present a GPS-based Computational Method of mean speed, which is aware of weather factor, road condition, and the instantaneous speed of road segment. In [23], Yamada et al. propose a method that prioritizes traffic log data in the basis of the contribution to prediction accuracy; each base station sends more important traffic log data to the server with higher priority. In [24], Ganapathy et al. propose a travel time based PrefixSpan (TT-PrefixSpan) algorithm which analyses traffic flow on highways by mining traffic sequence pattern and prediction of traffic volume based on traffic sequence rules. In [25], Zhang et al. propose a hybrid model to simultaneously predict the traffic flow in multiple positions by combining the layerwise structure and the Markov transition matrix (MTM), then they apply the methodology on the real-world traffic data from Xiamen city, China. In [26], Xie et al. propose a novel approach that combines the advantages of sequence-to-sequence (Seq2Seq) models and graph neural networks, which model traffic conditions on road networks is modeled as a sequential of graphs.

Many ITS systems are targeted to provide highly complete and accurate data of traffic flow. However, because of various reasons such as limited sensor coverage area, sensor failure or transmission error, it is common that some traffic flow data are lost. Many data mining methods were proposed

using a wide spectrum of techniques to estimate missing traffic data [27] such as k-nearest neighbors method [28], artificial neural networks [29], adaptive rolling smoothing (ARS) approach by dynamically tuning the filter parameters in a rolling horizon scheme for online applications [30], and analyzing spatio-temporal characteristics of traffic flow and the spatial location of road segments [31].

A prospective approach in ITS research is machine-learning-based vehicle detection over real-time images captured by existed camera systems. An Automated Vehicle Classification (AVC) system has three major modules or components, such as Features Extractor, Global Representation Generator, and Classifier [32]. Because the spreading of vision sensor systems around our city is too expensive, this approach does not solve the traffic problem completely. It just focuses on some particular points. Our methods improve traffic status observation in comparison with those deduced from vehicle classification (by YOLO model [33]) on reliable road images supplied by camera monitoring systems of HCMC Department of Transport. We will describe more details in Section "Urban Traffic Estimation from Crow-Sourced Data".

In Vietnam, there are Urban Traffic Programs provided by VoV (The Voice of Vietnam) and VoH (The Voice of Ho Chi Minh City People) [34, 35]. These systems encourage radio audiences to report traffic conditions, especially bad conditions such as heavy ones or congestion on their routines, via mobile phones. After being attentive to users' reports, reporters will broadcast such information on the radio channel with the idea that information she/he obtains is reliable. The foremost advantage of this mechanism comes from the benefits of the radio channel (for broadcasting the information) and mobile phone communication for reporting traffic data. However, there are lots of manual work-load for processing reported data to useful information about traffic status. Additionally, there's no valid mechanism to validate the reported data and take away noises. Therefore, to have effective traffic congestion monitoring and notifying system, two important issues must be resolved as follows:

- (i) Data must be collected completely (spatially and temporally) and accurately.
- (ii) The aforementioned data must be efficiently analyzed to estimate traffic status appropriately and timely.

Each of the prevailing studies and systems discussed above has its advantages and disadvantages. The foremost important advantages they have are utilizing mobile devices for increasing the coverage and applying crowdsourcing mechanisms for collected useful and informative data instead of just only GPS data. However, they continue to be various technical issues that require to be improved such as we want a straightforward yet efficient

framework to assist users easily collect and report data while conveniently retrieving traffic information and share the opinion (e.g., evaluating, scoring) on data reported by other users. To this end, two essential challenges must be resolved to lead the main differences between this work and literature as follows. We simplify the integration of multiple crowd-based data sources so that it efficiently fuses necessary data for traffic status estimation. Then novel data analysis mechanisms, including data validation, traffic estimation and prediction based on inference methods and machine learning approaches are proposed to process a large amount of data issuing accurate and update traffic status.

Overall Architecture and Problem Definition

This section presents the overall architecture of our proposed framework for urban traffic estimation from crowd-sourced data (UTECSO) and then describes the main problems to be resolved in this paper.

Overall Architecture

Figure 1 shows the overview of our proposed framework, namely the UTECSO, which contains four main components: data sources, data fusion and integration, traffic status analysis and prediction, and data warehouse. These components are briefly described as follows.

Data Sources

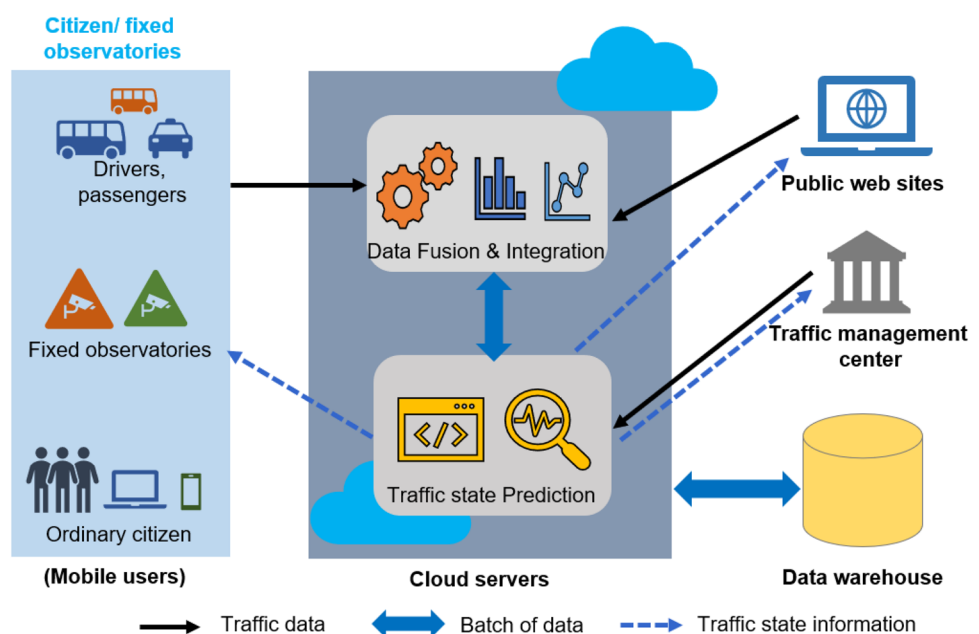
Traffic-related data are collected from many different sources, namely the crowd-sources such as:

- (i) From fixed monitoring systems (fixed observatories) which are already available like fixed camera-based for urban traffic monitoring systems [4];
- (ii) From public websites that publish traffic information, for examples the traffic portal of Ho Chi Minh City Department of Transport [4]), the radio channel for urban traffic information [34, 35], and so on;
- (iii) From mobile users (citizen observatories) who can collect and share traffic related data to the system for processing and analysis.

Data from (i) and (ii) have been available, our proposed system can access them through APIs (application programming interfaces) provided by the owners of these systems. This paper focuses on collecting and processing data from citizen observatories mentioned in (iii), and then resolving the essential issues of missing data to overcome the difficulty of estimating urban traffic real-time at any location (road segment).

The mobile crowd-sourced data can come from citizen who uses mobile phones with internet connection (mobile users). Mobile user can be driver, passenger on buses, taxis, people living along the main roads, especially in the areas where traffic jams often occur, or any ordinary people who wants to devote to the community via sharing data about traffic condition such as students, office staffs, pedestrians on the street, etc. To provide an effective way

Fig. 1 Overall architecture of the proposed UTECSO system



for mobile users collecting and sharing data, our proposed system provides a mobile application with a user-friendly interface where users can easily provide the necessary data with less effort. In the data collection process, many data is collected automatically using sensors already available in mobile phones such as GPS for location, moving velocity, clock for time and so on.

Data Fusion and Integration

Traffic data provided by citizen observatories will be integrated and pre-processed (e.g., data cleaning, noise removal, normalization,...) before being analyzed by a data analyzer in the center. Data cleansing techniques, eliminating redundancy, resolving missing data, validating the quality of the data shared by the crowds, etc., will be conducted in this component. Note that the data from the community often contains a lot of interference because there is no strict constraint compared to those in data collected from fixed sensor systems. Therefore, data validity and reprocessing are challenging issues in this research.

Traffic Status Analysis and Prediction

After the data are collected and gathered on the center, the traffic analysis and prediction component will analyze these data to estimate and predict the traffic status accurately and promptly. This traffic information is then displayed on the city's traffic map deployed on both the mobile application and a public website. There are many mechanisms for analyzing the collected data and transform them to appropriate traffic information.

This paper focuses on the two main classes of mechanisms: (i) the inference mechanism, and (ii) the prediction mechanism. The former one involves processing the mobile crowd data, identifying their impact factors (i.e., the importance between different data on calculation), and then bring them all into account for computing traffic condition at a considering road segment and at a given time. The later method resolve the problem of missing the real-time data. In that context, traffic condition in the considering road segment at a given time cannot be calculated directly from real time data (since there is no data reported at that time), this information should be predicted instead. To predict traffic condition at the locations where real-time data is missed, appropriate data mining techniques will be researched and proposed to analyze historical data, extracting the behavior of traffic conditions at the considering road segments at given time frames. This is one of the challenged research issues of this work.

Data Warehouse

To reduce the response time, we split the traffic data into different versions: a real-time database and a full database which is integrated as data warehouse. The real-time database consists of traffic status data which is collected in the given newest time frame (15 min), then it is calculated to deduct the velocity (to define traffic condition) of a road. On the other hand, the data warehouse stores all data which is collected from many data sources up to now. The full database is used for deducting the velocities (traffic condition) of roads where the current data is missed, as discussed in the previous subsection (traffic prediction). Because the real-time database's size is much smaller than the full database's one, the time displaying traffic status on electronic devices will be reduced significantly.

Problem Definition

As discussed above in the Section "[Overall Architecture](#)", to achieve an effective urban traffic estimation system, there are many issues need to be resolved. Among them, there are two essential issues this paper will focus and resolve as briefly described below:

1. How to effectively compute from the mobile crowd-sourced data, which commonly contains a lot of noise. Therefore, assessing traffic condition is a hard problem which is to be addressed in this paper. The proposed mechanism needs to be lightweight and appropriate to road segment level, hence increasing the accuracy as well as the usefulness of the estimated information.

2. The issue of missing data is still remaining even with data crowd-sourcing methods. It is necessary to propose an appropriate method for predicting traffic condition at locations where real-time data is missed.

Solutions for these issues are described in details in the next section.

Urban Traffic Estimation from Crowd-sourced Data

In this section, we propose solutions for the two essential issues discussed in Section "[Overall Architecture and Problem Definition](#)". We initially present our 'Crowd-Sourced Data Analysis for Traffic Estimation' method. Then, we will devise a novel solution for predicting traffic conditions at locations where real-time data is missed. This approach helps to resolve the issues of missing data and the limitation in system coverage, ultimately.

Urban Traffic Estimation from Mobile Crowd-Sourced Data

As discussed, one of the primary issues of the fixed-site camera-based traffic monitoring and estimation systems is the high deployment cost and coverage limitation. Obviously, it is difficult to deploy such systems to cover the whole transportation arrangement of the city so the necessity of gathering enough information has not been met. To overcome the coverage issue, we can use data acquisition solutions from the community by receiving information via calls such as the radio channel: VOV [34], VOH [35]. Meanwhile, gathering and processing of GPS signals from the cell phones (researched by HCMC University of Technology) [5] also helps to tackle this problem. Another solution is the utilization of the shared traffic data from the user's mobile application provided by the HCMC Department of Transport [4]. These are sensible and potential approaches when smart-phones become progressively popular and the expense of 3G, 4G, and wireless connection becomes increasingly less expensive.

Nevertheless, a significant issue in collecting data from the crowd is the trustworthiness of data which has not been resolved in existing studies. Therefore, it is essential to propose an appropriate data validation mechanism to validate the quality shared by the crowd, and thus the quality of traffic estimated information. Besides, users who share data are volunteers or who are interested in devoting to social evolution. There could be no strict constraints to assure that the data shared by the crowd is always correct in comparison with data collected by fixed sensor systems like camera systems.

In this work, we propose to enhance the data validity also by the supports of the crowd via their scoring on data shared by other users and their accumulative reputations obtained during the time they participate and contribute to the system. Concretely, the data shared by users whose reputation is high should be more trusted (reliable) compared to that shared by the user with a lower reputation. A user's reputation score will be calculated each period (e.g., every 5 min which is compliant with the period used for recomputing and updating traffic conditions). If a user has a positive action in our system (e.g., reporting estimated velocity with high accuracy), this user's reputation score will increase, and vice versa. The reputation score r_{i_p} of user u_i in period time p is calculated recursively by his/her reputation score in period time $p - 1$ and his/her contribution score c_{i_p} in period time p as shown in Eq. (1). This value ranges in $[0, 1]$ representing low to high reputation, respectively.

$$r_{i_p} = \min(r_{i_{p-1}} \times (1 + c_{i_p} \times r_{i_{p-1}}), 1) \quad (1)$$

where, c_{i_p} is the contribution score of user u_i in period time p . There two types of actions a user can contribute to the

system, namely sending a report of traffic data (e.g., the estimated velocity of traffic flow) and evaluating other users' reports. Both actions will change their reputation score, and it is worth noting that in a period of time p a user u_i can contribute by only one type of action: reporting or scoring, but not both. For each type, a user can conduct several actions. For example, a user u_i can report several data values to the system or can score for several other users' reports during a time period p .

If a user u_i reports a data point, namely X on a road segment S , for example, in the period time p , the contribution score c_{i_p} of user u_i is calculated in Eq. (2).

$$c_{i_p} = (S_{X_i} - r_{i_{p-1}}) \times (1 - r_{i_{p-1}}) \quad (2)$$

Here, the component $(S_{X_i} - r_{i_{p-1}})$ shows the change of user reputation score in the period time p , where S_{X_i} is the score of the user u_i receives from other users' evaluations, which is calculated in Eq. 3.

$$S_{X_i} = \frac{\sum_{j \in U_c} s_{ji} \times r_j}{k} \quad (3)$$

where,

- U_c is a set of k users who evaluate the report of user u_i ,
- $s_{ji} \in [0, 1]$ is the score that user u_j evaluates the data point X reported by user u_i ,
- $r_j \in [0, 1]$ is the reputation of user u_j .

If a user u_j evaluates a report X of user u_i , the c_{j_p} of user u_j at the period time p is calculated in Eq. (4)

$$c_{j_p} = (1 - |\text{median}(S_{ji})|) \times (1 - r_{j_{p-1}}) \quad (4)$$

where, $\text{median}(S_{ji})$ is the median of the scores evaluated by k evaluators mentioned above.

The validity V_{X_i} of the data X reported by user u_i is calculated in Eq. (5)

$$V_{X_i} = r_i \times S_{X_i} \quad (5)$$

Now, every data shared by the mobile crowd have been validated and ready for estimating traffic conditions. In this work, the traffic network is separated into road segments whose traffic conditions are estimated periodically, which is compliant with the period p used to update user reputation and data validation presented above. Each road segment is short enough so that traffic conditions at any location on a road segment can be considered the same. Thanks to Open Street Map [36], the concept of road segment has been implemented, we just modify some specific road segments (e.g., combining too short or dividing too long ones) to enhance the performance of our proposed system.

Traffic condition is estimated using the velocities v_{X_i} about traffic flows reported by mobile users. These velocities are integrated to obtain the estimated velocity of traffic flow during a time period of p (time frame). Let denote v_s the velocity of the traffic flow on the road segment s during the period p , it is estimated in Eq. (6).

$$v_s = \frac{\sum_{i \in U_s} v_{X_i} \times V_{X_i}}{\sum_{i \in U_s} V_{X_i}} \tag{6}$$

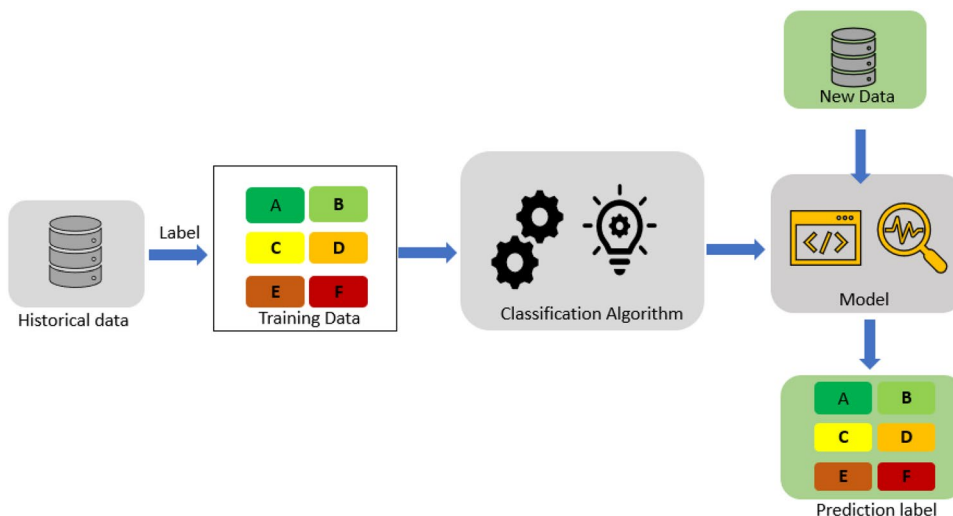
where U_s is the set of users who report data in the road segment s during the period p .

Then, this velocity (i.e., v_s) is mapped to the Levels of Service (LoS) whose values range from A to F representing traffic condition from “good” or “normal” to “congestion”, respectively, as presented in Table 1 [37]. These values are convenient to be displayed on the city map (on mobile or web-based application) using colors, such as from green to red, accordingly.

Table 1 Traffic levels of service in accordance with velocities

Level of service	Maximum velocity (40–50 km/h)	Maximum velocity (60 km/h)	Maximum velocity (70 km/h)
A	≥ 35	≥ 40	≥ 45
B	30–35	35–35	40–45
C	25–30	30–35	35–40
D	20–25	25–35	30–35
E	15–20	20–25	25–30
F	$V < 15$	$V < 20$	$V < 25$

Fig. 2 A functional prototype of the proposed system



Traffic Prediction at Missing Real-time Data Streets

Mining Urban Traffic at Locations Where Real-Time Data is Missed

Although the proposed system has many mechanisms to collect data from various sources (crowd-sources), the issue of missing real-time data remains. In this section, we propose novel approaches to mining traffic status information from historical data to solve this essential issue. There are two kinds of historical traffic data for each road segment, namely periodic data, and streaming data. The traffic periodic data are collected in a cycle of time, e.g. at rush hour on weekdays (at 8:00 AM each day). The traffic streaming data are data that is collected consecutively, e.g. last 30 min or one hour ago. Here, we combine both periodic and streaming data into the mining process where classification algorithms are utilized to estimate traffic conditions based on historical data. We proposed a model for training and estimating the traffic status at each segment in the specific timeline. The input of this model is the traffic condition of road segments at a given time, i.e., the data record is in the form (time_frame, roadSegment_ID, trafficCondition). Here, the traffic condition has been classified into LoS (LoS = A, B, C, D, E, and F) as discussed in the previous subsection shown in Table 1 [37].

These label-based data mentioned above are suitable for applying classification models in data mining techniques to “predict” the LoS/condition of road segments where real-time data are missed based on their behaviors hidden in the historical data. The proposed framework for traffic classification model is illustrated in Fig. 2 and is summarized as follows:

- *Step 1—Training data preparation* The purpose of this step is to conduct data pre-processing after collected

from many sources and then labeling them following the LoS. Concretely, the LoS can be calculated directly from velocity extracted from GPS data reported by individual vehicles, from other systems or it could be the output of this data mining model (ref. Step 3).

- *Step 2—Training the data classification model* As presented in Fig. 2, Classification models are trained by label-based historical data. The question here is that what classification model should be used so that it can be effectively trained by simple yet relevant LoS data, while the model itself is light-weight enough for best practices. According to this criterion, this work will adapt several models such as Decision Tree, Naive Bayes, Random Forest, Artificial Neural Network (ANN),... to find out the best-suited one.
- *Step 3—Applying the model to new data* The classification model built in Step 2 is applied to analyze the new coming data to determine the label/LoS of considering (missing real-time data) road segments.

With the general framework as mentioned above, the next step is how to determine the input data which is used to train the classification models effectively. There are some issues that we need to consider during finding the solutions. First of all, the traffic conditions may vary from road segment to the road segment even these segments are in the same street. Second, traffic conditions change frequently. To overcome these issues and provide appropriate traffic conditions in terms of space and time, we need to manage well the data by these two dimensions.

For the time dimension, we divide the time using time frames based on that the traffic conditions data is integrated and analyzed in the system. In the experiment, we choose the time frame as $\Delta t = 5$ min as it is short enough for averaging the traffic condition while it is long enough

for effectively computing traffic condition on the server [2].

For the space dimension, we divide a street into road segments based on the recommendation in Open Street Map [36] as presented in the previous subsection.

Mining Traffic Condition from Navigation Systems

As presented, besides the data collected from mobile users through an application on mobile devices, we also collect data from other systems and integrate them as historical data for the mining purpose. There are various existing navigation systems that are close to our proposed system where the structural traffic condition data can be efficiently crawled such as Google Map [38], Here maps [39], TomTom [40] and so on. After investigated, we figured out TomTom is one of the most convenient systems where we can collect traffic status using public API from them.

As displayed in Fig. 3, the fundamental components of the proposed traffic data collection system is categorized into 3 different groups as follows.

1. Application collects traffic status from TomTom's API running on local computers.
2. TomTom's API provides the traffic status of these specific streets selected to conduct the experiment.
3. Pre-processing and labeling the collected data with a consistent format for training and testing the proposed classification models in Fig. 2.

Mining Traffic Condition from Camera Systems

Besides, navigation systems, surveillance or traffic monitoring systems using cameras are potential crowd-sources which can be applied in our work. Fortunately, HCMC has

Fig. 3 Prototype of the proposed traffic status data collection system

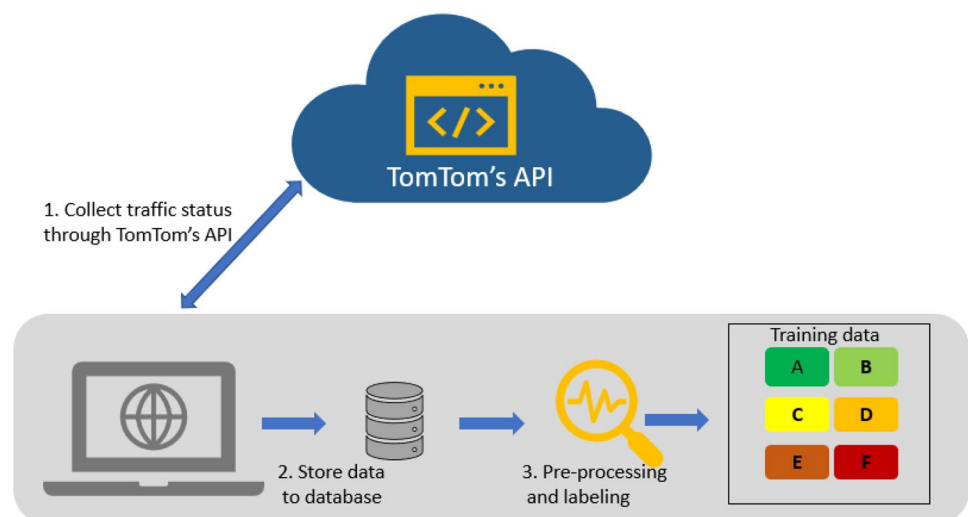
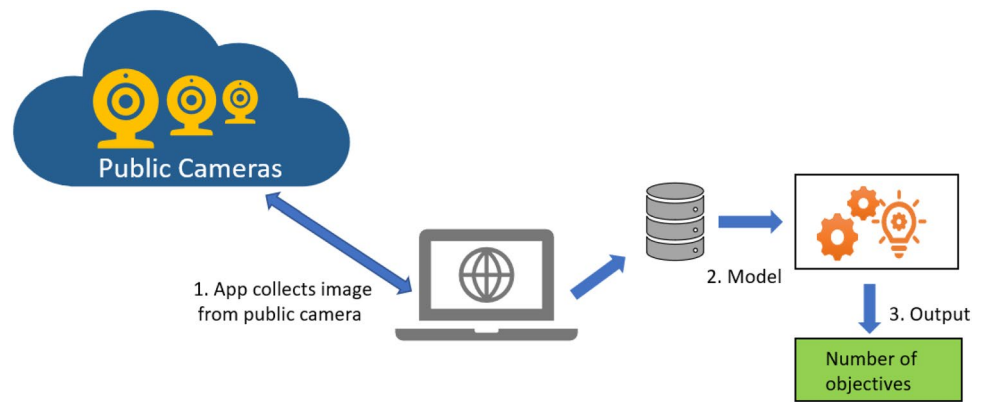


Fig. 4 Prototype of the image processing proposed system



deployed a camera system for traffic monitor by the Department of Transport [4] where all cameras are publicly accessible. Therefore, we use this system as another external source for collecting historical data to enrich the historical data source which supports for our proposed data mining method. The proposed method of using images from cameras to estimation the traffic status is presented in Fig. 4 and is summarized as follows:

1. Application collects images about traffic from cameras running on local computers using Python with the time frame (5 min).
2. Using a model to count objects in the images. In this step, images are analyzed to detect and count the number of vehicles contained in them using a YOLO v3 model (describe in below).
3. Prediction the level of services based on the number of objectives. This step predicts the traffic status based on the number of vehicles in the images which are estimated in Step 2.

For further information about the model used in Step 2, we used the YOLOv3 model [33] (or You Only Look Once v3, by University of Washington), as it is a popular object detection method which is recognized as one of the fastest object detection algorithms in the practice. It is a good choice when we need real-time detection, without loss of too much accuracy [41], which is compliant with our work.

The prototype of the proposed system is presented in detailed in the next section.

Prototyping

To properly illustrate the validity of the proposed approaches, a prototype system that addresses all the previously utilize cases through dedicated interfaces and sub-systems was deployed. The overview of this prototype is presented in Fig. 5. The elemental components of the system

prototype are categorized into four different groups, mostly according to their physical location and functionality:

1. Client applications (mobile and web-based applications) used the data collection, sharing as well as accessing traffic information from the system.
2. Business processing module that processes data to estimate traffic conditions.
3. Database management system to store all data.
4. External systems used as other data sources.

More details about the functionality of each component are presented in the next paragraphs.

Client Applications

User data from the community are vital in a crowd-sourcing application. Keeping the number of users and the accuracy of shared information is crucial. These factors must be guaranteed when building the system of traffic data gathered from users. Mobile and web applications are two fundamental correspondence channels between users and the system which allows users to share traffic conditions and check it on the traffic map in the applications.

Figure 6 shows the graphical user interface (GUI) on the mobile application (android) when a user reports traffic status from a street. On the left side, a form with a consistent format appears to ask the user to fill the necessary values before sending the report to the system. The middle part demonstrates a dynamic routing when the user enters a destination where they want to go to. The system will get the current location of the users based on the GPS signal and then ask the confirmation from the user to check if the current location is correct. Based on a pair of locations (current location, destination) and the current traffic situation, the system will compute to find the best routing at that time for user and displays that route on the map. The right side shows that the system also helps the user to confirm the direction (on the road) to which the user is going to report. We have

Fig. 5 A functional prototype of the proposed system

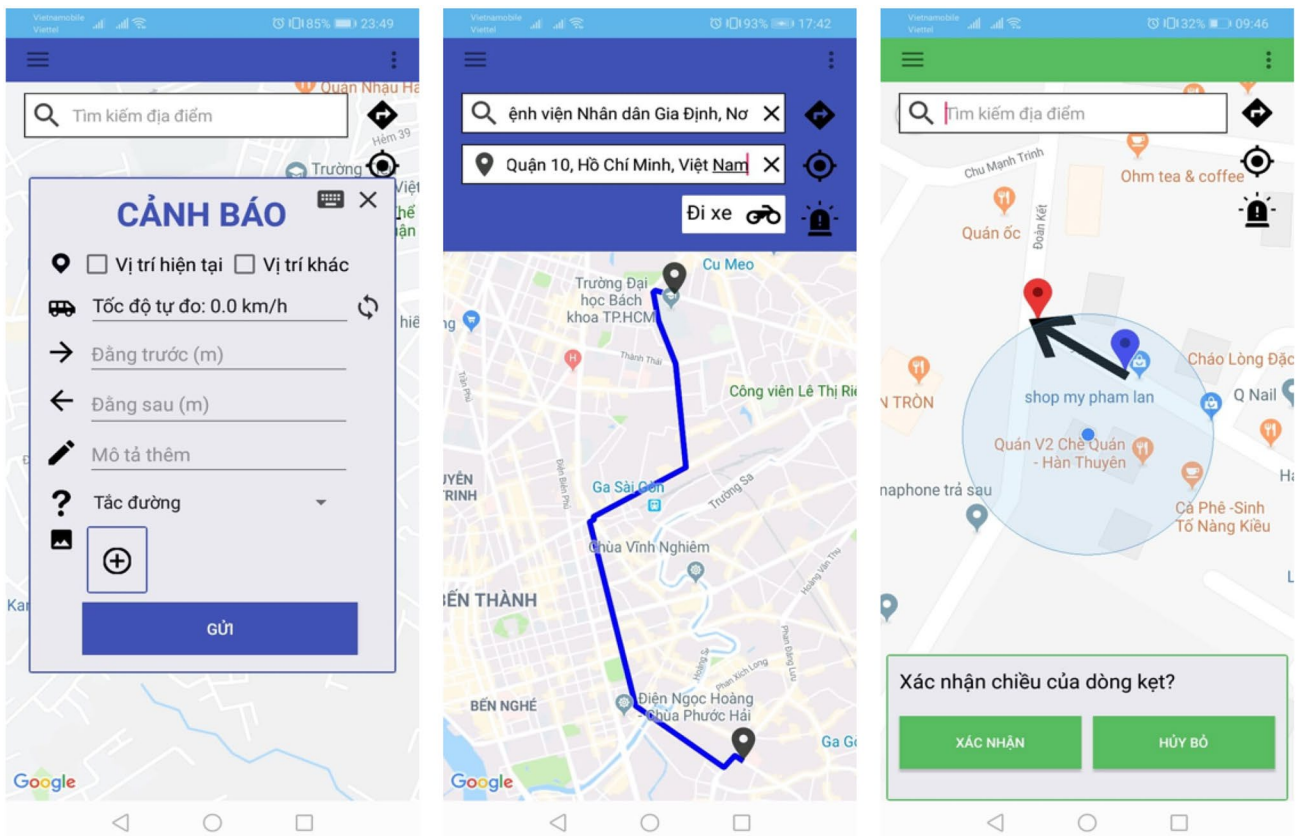
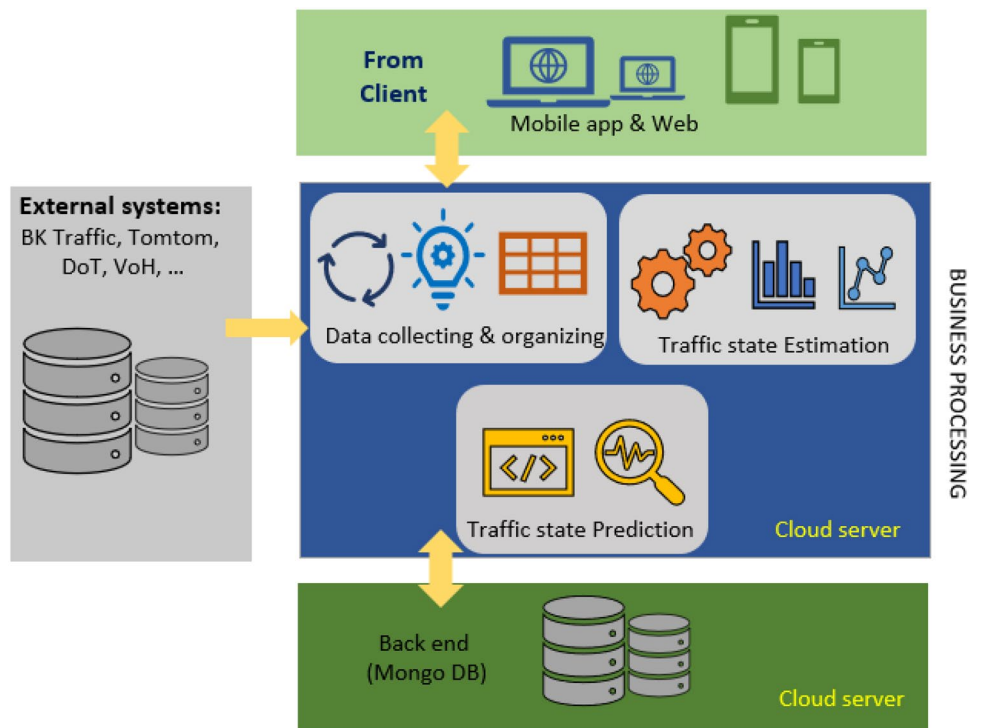


Fig. 6 Mobile application for end users

already published this application to allow users to install and contribute traffic data.

Business Processing Module

This component consists of three components: (i) Data Collecting and Organizing, (ii) Traffic Status Estimation, and (iii) Traffic Status Prediction.

(i) Data Collecting and Organizing

collects and organizes data from mobile applications, webs, as well as external data sources.

Information about traffic condition from users includes:

- Location: to identify the location of the heavy traffic
- Speed: is the average speed of the traffic flow estimated by users
- Head and tail: to identify the length of traffic flows

In addition, there are some other data such as Comments, Reasons and Images

This data assists with recognizing traffic conditions all the more precisely and obviously. So as to get traffic conditions from publicity, a few APIs have been built as follows.

Account information, and authentication API:

- Register account
- Login to the system
- Login via third-party
- Retrieve account information
- Update user account
- To get traffic condition in a specific area

(ii) Traffic Status Estimation

performs velocity computations and updates user reputation points. The server will fetch the crowd-sourcing data got from end-users in a time frame that stored in the database server for calculation, then updates the velocity and user reputation points to the Database server. The calculation is based on the recipes given in Section "Urban Traffic Estimation from Crowd-Sourced Data". Updating the velocity of the segment is split into a cycle (called a time frame). Whenever a cycle starts, the server begins to retrieve the crowd-sourcing data of this cycle. After the calculation is finished, the server checks whether the segment has not been updated for speed yet. If not the server will call through the Resource APIs to update these segments.

Traffic Data Sharing API:

- Share traffic status while moving
- Share traffic status remotely

- Retrieve current shared information data
- Get shared traffic condition based on time

(iii) Traffic Status Prediction

plays a critical part in case crowd-sourced data is missed. At this time the system requires the function of anticipating traffic conditions utilizing data mining techniques presented in Section "Urban Traffic Estimation from Crowd-Sourced Data".

As presented in Section "Urban Traffic Estimation from Crowd-Sourced Data", we proposed the 'mining traffic condition from navigation systems' and 'mining traffic condition from camera systems' for traffic prediction at missing real-time data streets. In this subsection, we have deployed a prototype system for those.

Prototype of mining traffic condition from TomTom system

An application is built and deployed to collect the traffic status from TomTom running on local computers. We conducted collecting traffic status at specific streets in HCMC, namely TruongChinh (TC), HoangVanThu (HVT), CongHoa (CH) as shown in Fig. 7 through public APIs from TomTom. These streets are selected for collecting traffic status because there is usually heavy traffic congestion.

The data format getting from TomTom is $\langle \text{longlat}, \text{timestamp}, \text{velocity} \rangle$. However, in order to display the traffic conditions on Open Street Map [36], we need segmentID and LoS instead of (long lat and velocity). Therefore, we need to have a mechanism to map from long, lat to segmentID and convert velocity to LoS as shown in Table 1.

For having the expected training dataset including 3 following dimensions that are relevant for the mining task: $\langle \text{segmentID}, \text{timestamp}, \text{LoS} \rangle$, we have conducted an API for pre-processing by using Python.

Based on the basic data collected from TomTom, we combine several parameters to make the training dataset more effective by following a technique applied in this work.

- *Technique 1* We gather all historical data in the same road segment, the same time frame, and the same day of the week. This one is combined with a training dataset. For example, with segment A belongs to the street B at time frame 8:00–8:05 in every Tuesday morning has a training dataset, shown in Table 2.
- *Technique 2* We gather all historical data in the same street, in the same time frame and the same day of the week (the set of all segments on a street) are used to train the classification model. For example, the street B at time frame 8:00–8:05 in every Friday morning has a training dataset as the following Table 3.



Fig. 7 The map of streets collecting the traffic status

Table 2 Input data for training the proposed approach with Technique 1

Segment Id	Timestamp	LoS	Weekday	Tuesday
42446	7:35:00	D	Yes	Yes
42446	7:35:00	D	Yes	Yes
42446	7:35:00	C	Yes	Yes
42446	7:35:00	E	Yes	Yes
...
42446	7:35:00	E	Yes	Yes

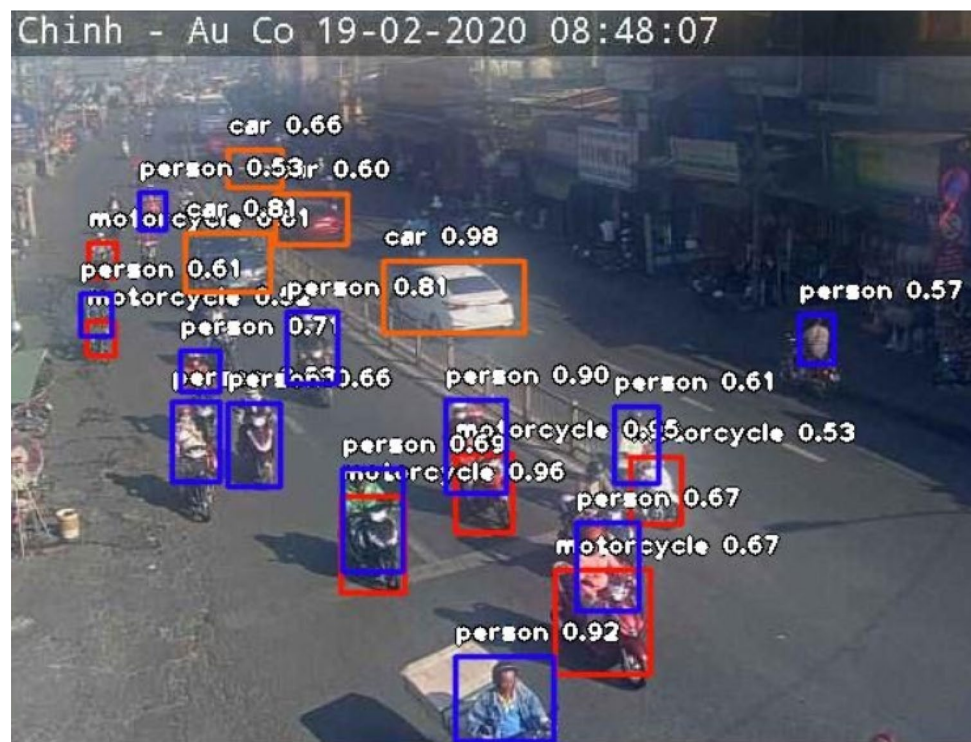
Table 3 Input data for training the proposed approach with Technique 2

Segment Id	Timestamp	LoS	Weekday	Friday
42446	7:35:00	B	Yes	Yes
42447	7:35:00	D	Yes	Yes
42448	7:35:00	C	Yes	Yes
42449	7:35:00	A	Yes	Yes
...
42459	7:35:00	E	Yes	Yes

Prototype of mining traffic condition from camera systems

As presented in Fig. 4 in Section "Urban Traffic Estimation from Crowd-Sourced Data", we have built an experimental system to collect the traffic status from public cameras then analyze and predict traffic conditions based on historical images. Currently, the prototype system is experimenting on a local computer with the following steps. First, the images about traffic are collected using Python application on the cameras at these streets TC, HVT, CH every 5 min. All images data is stored into a folder on a local computer with a consistent format as StreetName-Direction-Timestamp, for instance, HoangVanThu-D1-190220200740. Second, we use the YOLOv3 model to detect how many objects are there in the picture including, person, motorbike, car, and bus then output with a number. Third, based on the result in the second step, we output a specific number in this picture, namely D (density). Nevertheless, we need to have a mechanism to predict and estimate the traffic status based on D, unfortunately, we have not finished this functional yet. This one is investigating as well as finding the appropriate method to evaluate the traffic conditions rely

Fig. 8 Estimation traffic status by image processing



on D. Therefore, we would like to take off it for subsequent expanded considers of the system inside the near future. The experimental results from step 2 are quite interesting and potential at the beginning, not only fast but also acceptable accuracy. Figure 8 is a result after used the model to detect the objects.

Database Management System to Store All Data

There are two databases in the system: one is for storing traffic conditions receiving from the client via the VoH channel, and the other is for storing the data that is computed. These are named “VoH” and “crowd-sourcing”, respectively. To adapt with a real-time service, our system needs to have these below characteristics:

- Ability to retrieve and process large data in a short time
- Multiple formats as images, text,...

The crowd-sourcing database is used to store the traffic statuses that were collected from the end-user via client application, whereas the VoH database stores information of traffic updated from broadcasters throughout VoH channels. These are located at the Ho Chi Minh University of Technology

We have chosen MongoDB to implement for the Database management system, because, MongoDB is open-source with these advantages such as High performance, Flexible data, Scalability, Rich query language.

External Systems Used as Other Data Sources

In addition to collecting data from the end-users, the system also gathers, exploits, and uses the data of traffic status from external systems such as BK Traffic system, VoH system, Department of Transport system [4, 5, 34].

Evaluation

In this section, we discuss and evaluate the feasibility, effectiveness and the efficiency of the proposed approach.

Evaluation Environment

We conducted experiments using the implemented prototype system presented in Section “Prototyping”. Besides, evaluating the feasibility of the system on its functionality flows (i.e., satisfying user requirements on sharing traffic data and accessing traffic information), we also evaluate the response time when there various concurrent requests. Response time of an API (i.e., a function of the system) is the sum of sent time, executed time, and received time of this API. We use Jmeter [42]—a popular open-sourced tools - to implement this evaluation. The server’s hardware configuration consists of Intel Xeon E-2640 (6 cores, 2.5GHz), RAM 32GB, Bandwidth 14 Mbps. A client hardware configuration is Intel

Table 4 Response time of eight APIs with various throughputs

API	Response time (ms)	Throughput (request/s)
/login(post)	245	45
/my(get)	326	50
/my(post)	224	40
/traffic-reports(get)	353	17
/traffic-reports/move(post)	695	13
/traffic-reports/remote(post)	303	36
/evaluation(post)	374	49
/traffic-status(get)	365	40
Average time	360.5	36.25

Core i3 (2 core, 1.7 GHz), RAM 4 GB. Mobile users also use Android smartphones to install our implemented mobile application for testing the system.

Feasibility

First, experiments on using our developed mobile application for collecting, sharing traffic data and accessing traffic information from the system have been conducted. The results reveal that system's functions are correctly work in accordance with our system analysis and designs.

Second, we have evaluated the response times of main functions using the environment presented in the previous subsection. Table 4 shows the result of the eight main functions, where the response time of the heaviest functions is around 600ms with the throughput is about 15 requests per second. These indicators reveal that the proposed system is feasible with current technology.

Effectiveness

In this subsection, we present and discuss the effectiveness of the proposed classification approach to mine traffic information at location where real-time data is missed. Here, Precision, Recall, and F-Score are used as indicators of the evaluation and they are described as follows.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (8)$$

$$F\text{score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (9)$$

where TP, FP, and FN are true positive, false positive and false negative, respectively.

We have conducted the experiments in 2 stages: (i) using the GPS data from the bus system in HCMC [5], and (ii) using data collected by our prototype system presented in Section "Urban Traffic Estimation from Crowd-Sourced Data".

Stage 1: Evaluation Using GPS Data from City Bus System

We applied the proposed model to analyze the traffic condition extracted from GPS data provided by buses in HCMC [5] which is described as follows.

- The format of this data is presented in Table 5 with 6 dimensions, namely SegmentID, Speed(LoS), Time, Frame, Weekday, Peak_time.
- Data mining techniques were applied are Decision Tree, Random Forest, ANN, Support Vector Machine.

Among 4 classification models mentioned above, Random Forest is prominent with Recall = 47%, Precision = 44% and F-score = 45%. This result reveals that the proposed classification model is able to predict the traffic condition of missing data road segments. However, the performance (i.e., the recall, precision, and F-score) is still low to satisfy the practical requirement. This drawback is because the traffic data from buses might not be enough and be limited in coverage as there are many streets without buses. That is while our proposed prototype system described in Section "Urban Traffic Estimation from Crowd-Sourced Data" is useful for concentrating on the collection of real-field traffic data at considering road segments.

Stage 2: Evaluation Using Real-Field Data Collected by Our Prototype System

As mentioned above, we used an application on local computers to automatically collect the traffic-related data via TomTom [40] for 3 weeks, resulted in 39,685 valid records obtained.

In this stage, the Random Forest algorithm is applied too because it also provides the highest performance as stage 1,

Table 5 Data from buses provided by BK traffic [5]

SegmentID	Speed	Timestamp	Frame	Weekday	Peak_time
9549185	28.5	1527807556745	23	4	0
9549185	32.25	1527809356997	25	4	1
9549185	28.5	1527827357076	43	4	0
9549185	21.67	1527836357362	55	4	1

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.764	0.117	0.766	0.764	0.765	0.647	0.915	0.857	Dong Di Chuyen Duoc
0.671	0.073	0.671	0.671	0.671	0.598	0.893	0.783	Dong Di Chuyen Binh Thuong
0.664	0.015	0.746	0.664	0.703	0.685	0.973	0.804	Thong Thoang
0.871	0.105	0.852	0.871	0.861	0.763	0.953	0.905	Dong Di Chuyen Cham
0.708	0.002	0.810	0.708	0.756	0.755	0.998	0.888	Un Tac
0.784	0.097	0.783	0.784	0.783	0.689	0.931	0.860	

Fig. 9 The result of historical data using classification model

and the input data is shown in Tables 2, 3 in Section "Urban Traffic Estimation from Crowd-Sourced Data".

The result is quite interesting as shown in Fig. 9 where Precision, Recall and F-Score are much greater than those in the previous one (stage 1), as shown in Table 6.

Stage 2 provides better results as all performance indicators increase around 33% compared to those in stage 1 using the same algorithm, namely Random Forest. Moreover, the number of dimensions in the input data used for the mining tasks in stage 2 is also lower (4 compared to 6). This means that it is more convenient to prepare the training data for the proposed mining approach. The results reveal the effectiveness of the proposed approach and thus it is feasible to be applied in the practices with existing technologies (i.e., mobile and cloud computing).

Conclusion

This paper proposes novel approaches to mobile crowd-sourcing based traffic condition estimation. We devised a framework where mobile crowd-sourced data is effectively collected, integrated, and processed for estimating traffic condition accurately and timely. In addition, essential issues on missing data in a large coverage systems, which remain as critical challenges, have been resolved by data mining approaches applied to the historical data. A prototype system has been developed to validate the feasibility and the effectiveness of the proposed approaches. Experimental results reveal that the proposed solutions are effective and ready to be applied in the practice. However, a lot of problems need to be tackled to increase the number of users who can share useful traffic data in the city. More research and approaches to incentive models should be conducted.

In addition, further features such as other traffic condition warnings (flooding, restricted streets etc), real-time response of traffic condition, and so on can be developed to increase the values of this research. These are potential research directions for our future work.

Acknowledgements This research is funded by the Department of Science and Technology (DoST), Ho Chi Minh City, under grant number 34/2018/HD-QKHCN.

Table 6 The performance of classification methods with data collected by our proposed system compared to that with data from other system

Method	ReCall (%)	F-score (%)	Precision (%)	Algorithm
Stage 1	47	45	45	Random Forest
Stage 2	78.	78.3	78.3	Random Forest

Compliance with Ethical Standards

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

- Quang T, Baharudin M, Kamioka E. Mc-tes: an efficient mobile phone based context-aware traffic state estimation framework. *J Inf Process.* 2013;21(1):76–9.
- Minh QT, Pham-Nguyen HN, Tan HM, Long NX. Traffic congestion estimation based on crowd-sourced data. In: *Proceedings of the 2019 International Conference on Advanced Computing and Applications (ACOMP)*. 2019. pp. 119–126.
- Quang T, Baharudin M, Kamioka E. Synergistic approaches to mobile intelligent transportation systems considering low penetration rate. *Elsevier PMC.* 2014;10(Part B):187–202.
- D. of Traffic and Transportation. Road traffic portal, Ho Chi Minh City. <http://giaothong.hochiminhcity.gov.vn/>, 2019. Online; Accessed Apr 2019.
- Vu P T. Smart bk traffic. <http://traffic.hcmut.edu.vn/index.html>, 2019. Online; Accessed Apr 2019.
- Ndoye M, Totten V, Krogmeier J, Bullock D. Sensing and signal processing for vehicle reidentification and travel time estimation. *IEEE Trans Intell Transp Syst.* 2011;12(1):119–31.
- J. M. of Land Infrastructure and Transport. The system outline of vics. <https://www.vics.or.jp/en/index.html>, 2019. Online; Accessed May 2019.
- Estellés-Arolas E, González-Ladrón-de Guevara F. Towards an integrated crowdsourcing definition. *J Inf Sci.* 2012;38(2):189–200.
- Lease M, Yilmaz E. Crowdsourcing for information retrieval. *SIGIR Forum.* 2012;45(2):66–75.
- See L, Mooney P, Foody G, Bastin L, Comber A, Estima J, Fritz S, Kerle N, Jiang B, Laakso M, Liu H-Y, Milšinski G, Nikšič M, Painho M, Pödör A, Olteanu-Raimond A-M, Rutzinger M. Crowdsourcing, citizen science or volunteered geographic information? the current state of crowdsourced geographic information. *ISPRS Int J Geo-Inf.* 2016;5(55).

11. Deanne B, Katharine H, Megan L, James OB. The use of crowd sourcing for gathering information about natural disasters. *Risk Fronter*. 2011;11(2):1–4.
12. Ushahidi, Ushahidi, read the crowd. <http://www.usahidi.com>, 2018. Online; Accessed May 2019.
13. Nguyen D-B, Dow C-R, Hwang S-F. An efficient traffic congestion monitoring system on internet of vehicles. *Wirel Commun Mob Comput*. 2018;2018:1–17.
14. Lewandowski M, Płaczek B, Bernas M, Szymała P. Road traffic monitoring system based on mobile devices and bluetooth low energy beacons. *Wirel Commun Mob Comput*. 2018;2018.
15. Aissaoui R, Menour H, Dhraief A, Filali F, Belghith A, Abu-Dayya A. Advanced real-time traffic monitoring system based on v2x communications. In: 2014 IEEE International Conference on Communications (ICC), 2014. pp. 2713–2718.
16. Elloumi M, Dhaou R, Escrig B, Idoudi H, Saidane LA. Monitoring road traffic with a uav-based system. In: 2018 IEEE Wireless Communications and Networking Conference (WCNC), 2018. pp. 1–6.
17. de Almeida TT, Nacif JAM, Bhering FP, Júnior JGR. Doctrans: a decentralized and offline community-based traffic monitoring system. *IEEE Trans Intell Transpo Syst*. 2019;20(3):1160–9.
18. Pham DT, Hoang BAM, Thanh SN, Nguyen H, Duong V. A constructive intelligent transportation system for urban traffic network in developing countries via gps data from multiple transportation modes. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, 2015. pp. 1729–1734.
19. Bayen A. Mobile millennium project. <http://traffic.berkeley.edu/>, 2019. Online; Accessed May 2019.
20. Zhao Y, Zhang Y, Yu T, Liu T, Wang X, Tian X, Liu X. Citydrive: A map-generating and speed-optimizing driving system. In: IEEE INFOCOM 2014—IEEE Conference on Computer Communications, 2014. pp. 1986–1994.
21. Zhao Y, Li S, Hu S, Su L, Yao S, Shao H, Wang H, Abdelzaher T. Greendrive: A smartphone-based intelligent speed adaptation system with real-time traffic signal prediction. In: 2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS), 2017. pp. 229–238.
22. Ma Q, Zou Z, Ullah S. An approach to urban traffic condition estimation by aggregating gps data. *Clus Comput*. 2019;22:5421–34.
23. Yamada Y, Shinkuma R, Sato T, Oki E. Feature-selection based data prioritization in mobile traffic prediction using machine learning. In: 2018 IEEE Global Communications Conference (GLOBECOM), 2018. pp. 1–6.
24. Ganapathy J, Paramasivam J. Prediction of traffic volume by mining traffic sequences using travel time based prefixspan. *IET Intell Transp Syst*. 2019;13(7):1199–210.
25. Zhang S, Kang Z, Zhang Z, Lin C, Wang C, Li J. A hybrid model for forecasting traffic flow: Using layerwise structure and markov transition matrix. *IEEE Access*. 2019;7:26002–12.
26. Xie Z, Weifeng L, Huang S, Lu Z, Du B, Huang R. Sequential graph neural network for urban road traffic speed prediction. *IEEE Access*; 2019.
27. Li L, Zhang J, Yang F, Ran B. Robust and flexible strategy for missing data imputation in intelligent transportation system. *IET Intell Transp Syst*. 2018;12(2):151–7.
28. Sun B, Ma L, Cheng W, Wen W, Goswami P, Bai G. An improved k-nearest neighbours method for traffic time series imputation. In: 2017 Chinese Automation Congress (CAC), 2017. pp. 7346–7351.
29. Pamula T. Impact of data loss for prediction of traffic flow on an urban road using neural networks. *IEEE Trans Intell Transp Syst*. 2019;20(3):1000–9.
30. Chen X, Zhang S, Li L, Li L. Adaptive rolling smoothing with heterogeneous data for traffic state estimation and prediction. *IEEE Trans Intell Transp Syst*. 2019;20(4):1247–58.
31. Liu A, Li C, Yue W, Zhou X. Real-time traffic prediction: A novel imputation optimization algorithm with missing data. In: 2018 IEEE Global Communications Conference (GLOBECOM), 2018. pp. 1–7.
32. Boukerche A, Siddiqui AJ, Mammeri A. Automated vehicle detection and classification: models, methods, and techniques. *ACM Comput Surv*. 2017;50(5):1–39.
33. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. pp. 779–788.
34. VoV, Road traffic channel fm 91mhz. <http://vovgiaothong.vn/>, 2019. Online; Accessed May 2019.
35. VoH, Urban traffic channel fm 95.6mhz. <http://voh.com.vn/thong-bao/kenh-giao-thong-do-thi-tren-lan-song-fm-95-6-mhz-cua-dai-tndd-tphcm-111909.html>, 2019. Online; Accessed May 2019.
36. Openstreetmap. <https://www.openstreetmap.org>, 2019. Online; Accessed Feb. 2020.
37. M. of Transport of Vietnam. Table of speed specifications for all drivers [online]. <https://luatvietnam.vn/tin-phap-luat/bang-quy-dinh-ve-tocto-toi-da-moi-tai-xe-can-nho-230-18150-article.html>, 2018. Online; Accessed Apr. 2019.
38. Google map. <https://www.google.com/maps/@10.7984228,106.656438,15.51z/data=!5m1!1e1>, 2019. Online; Accessed March. 2020.
39. Here maps. <https://wego.here.com/traffic/explore?map=10.79833,106.65405,16,traffic>, 2019. Online; Accessed Mar 2020.
40. TomTom I. Tomtom, inc. <https://developer.tomtom.com/products/traffic-api>. Online; Accessed Feb. 2020.
41. Kathuria A. What's new in yolo v3. <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b>, 2020. Online; Accessed Feb 2020.
42. JMeter A. Apache jmeter. <https://jmeter.apache.org/>. Online; Accessed Jun. 2019.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.